

UNITED STATES PATENT APPLICATION
OF
HSIEN-CHUNG WOO
FOR
AUTOMATIC SWITCHOVER MECHANISM IN A NETWORK DEVICE

AUTOMATIC SWITCHOVER MECHANISM IN A NETWORK DEVICE

BACKGROUND OF THE INVENTION

[0001] In a conventional computer network (e.g., a local area network or wide area network), computers communicate over a network infrastructure made up of interconnected nodes, such as routers and/or switches, connected by communication links such as optical fiber, copper cable, and wireless links. Often, the topology is such that each node connects to multiple neighboring nodes (e.g., as in a mesh configuration).

[0002] Typically, data propagates through such a network in discrete units referred herein as “packets” (meaning any fixed- or variable-sized discrete unit of data, including frames, cells, segments, or packets). For each received packet, a node examines the contents of the packet (e.g., a destination address in the packet header) and determines which neighboring node to forward the packet to. Often, a node makes this determination by performing a table lookup. The neighboring node performs a similar determination until the packet eventually propagates from node to node through the network to its destination.

[0003] When a node (or a component in the node) fails, the node may no longer be operative to forward packets. Such a failure may result in serious disruption to the performance of the network. Even after the failure is repaired, there may be a delay before the node becomes fully operational, resulting in continued disruption to the network. In routers, for example, the operation of certain components may depend upon state information, which is maintained in the router according to the contents of previously forwarded packets. Accordingly, once the failure in the router is finally repaired, it may take some amount of time for the router to generate sufficient state information for it to function properly.

[0004] Thus, there is a need for an invention that more adequately addresses such problems which may occur in the network.

SUMMARY OF THE INVENTION

[0005] Additional objects and advantages of the invention will be set forth in part in the description which follows, and in part will be obvious from the description, or may be learned by practice of the invention. The objects and advantages of the invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims.

[0006] In accordance with the principles of the invention, a system comprises a plurality of processing modules, one of which is designated to be the primary processing module and the others are designated to be secondary processing modules. During operation, state is maintained in the primary processing module and at least one of the secondary processing modules. A switchover controller causes outputs from the secondary modules to be discarded. When the switchover controller receives an indication that the primary processing module has failed, it designates one of the secondary processing modules to be the primary processing module. Because the newly designated primary processing module already has current state information at switchover, the module is able to operate with minimal delay.

[0007] It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory only and are not restrictive of the invention, as claimed.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate one (several) embodiment(s) of the invention and together with the description, serve to explain the principles of the invention.

[0009] Fig. 1 is a block diagram of a first system incorporating principles of the invention;

[0010] Figs. 2A and 2B are block diagrams of interface modules usable in the system shown in Fig. 1;

[0011] Fig. 3 is a flow chart of a first process incorporating principles of the invention;

[0012] Fig. 4 is a block diagram of a second system incorporating principles of the invention;

[0013] Fig. 5 is a flow chart of a second process incorporating principles of the invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0014] According to the invention, a system comprises a plurality of processing modules, one of which is designated to be the primary processing module and the others are designated to be secondary processing modules. During operation, state information is maintained in the primary processing module and at least one of the secondary processing modules. The processing modules may obtain the state information from a received data stream (e.g., information contained in packet headers) or received control information (e.g. routing engine configuration information or control traffic). A switchover controller causes outputs from the secondary modules to be discarded. When the switchover controller receives an indication that the primary processing module has failed, it designates one of the secondary processing modules to be the primary processing module. Because the newly designated primary processing module already has current state information at switchover, the module is able to operate with minimal delay. The invention is described herein in connection with a network switching device, but may be usable in other types of devices that use modules in which state information is maintained.

[0015] Fig. 1 shows system 100 which incorporates principles of the invention. As shown in Fig. 1, system 100 includes routing engine 110, forwarding planes 120A and 120B, interface modules 130, and switchover controller 140. Routing engine 110 is coupled to both forwarding planes 120A and 120B. Each interface module 130 is coupled to both forwarding planes 120A and 120B. Switchover controller 140 is coupled to forwarding planes 120A and 120B and each of the interfaces 130. System 100 is preferably a network switching device (e.g.,

a router or switch), which interconnects with other switching devices to form part of a network, and accordingly may include other components used in a network switching device.

[0016] Routing engine 110 is responsible for generating routing and/or forwarding tables based on information it systematically receives about the network in which system 100 is connected. Such information is typically received using conventional routing protocols. After routing engine 110 generates a forwarding table, it transmits the forwarding to each of the forwarding planes 120A and 120B for use in forwarding packets. In a preferred embodiment, routing engine 110 is implemented as a processor executing software modules.

[0017] Forwarding planes 120A and 120B handle packet forwarding. Each forwarding plane includes a plurality of inputs at which data packets are received and a plurality of outputs from which data packets are transmitted. In one embodiment, each forwarding plane is a packet forwarding engine, configured to analyze destination information contained in the headers of received packets and, using a forwarding table, determine appropriate outputs for transmitting the received packets. In another embodiment, each forwarding plane is a TDM (time division multiplexed) plane, configured to analyze destination information contained in received packets and transmit the received packets onto channels at outputs based on appropriate time slots. Forwarding planes 120A and 120B can be implemented in hardware using ASICs or other circuit technology, or in software executing modules to perform the appropriate functionality, or in a combination of hardware and software. The determination of the appropriate outputs for respective packets is preferably carried out using a processor configured to perform table lookups. While two forwarding planes are shown in Fig. 1, more than two forwarding planes could also be used. In accordance with one embodiment of the invention, both forwarding planes receive and process the same packets from interface modules 130 (so that they can both maintain the same state information), though only packets forwarded by the primary forwarding plane are output by interface modules 130.

[0018] Interface modules 130 connect to respective external lines and forwarding planes 120A and 120B to transmit packets between the external lines and forwarding planes 120A and 120B. In receiving packets from the external lines, interface modules 130 preferably perform some processing on the received packets (e.g., layer 2/3 header processing) and then transmit the packets to forwarding planes 120A and/or 120B. In transmitting packets out on the external lines, interface modules 130 preferably perform some processing from packets received from forwarding planes 120A and/or 120B and then transmit the packets to the respective lines. In a preferred embodiment, interface modules 130 are equipped to handle characteristics unique to the external lines to which they are connected. For example, if the external lines are fiber optic, interface modules 130 may incorporate optical components, such as lasers, to receive and transmit packets over the fiber optic lines. While two interface modules 130 are shown, fewer or more interface modules 130 may be used. In addition, while each interface module 130 is shown as having a single incoming line and a single outgoing line, interface modules 130 could have more incoming and outgoing lines.

[0019] Figs. 2A and 2B show different embodiments of interface module 130. In Fig. 2A, interface module 130 includes spray logic 210 and despray logic 220. Spray logic 210 receives packets at an incoming line and, under the control of a signal from the switchover controller, outputs the received packets to either forwarding plane 120A or forwarding plane 120B. Despray logic 220 receives packets from both forwarding plane 120A and forwarding plane 120B and, under the control of a signal from the switchover controller, outputs the received packets from one of the forwarding planes to an outgoing line. In one embodiment, spray logic 210 could be implemented as a demultiplexor and despray logic 220 could be implemented as a multiplexor.

[0020] Fig. 2B shows another embodiment of interface module 130. As shown in Fig. 2B, interface module 130 includes duplicator logic 230 and despray logic 240. Duplicator logic 230 differs from spray logic 210 in that duplicator logic 230 duplicates received packets for

transmission to both forwarding planes 120A and 120B, while spray logic 210 transmits received packets to only one of the forwarding planes. Because duplicator logic 230 transmits received packets to both forwarding planes, it is shown as not receiving a signal from switchover controller 140. In embodiments where interface module 130 connects to more than two forwarding planes, duplicator logic 230 could receive a signal from switchover controller 140 to transmit received packets to both a primary forwarding plane and a single secondary forwarding plane. Despray logic 240 is substantially similar to despray logic 220.

[0021] Referring back to Fig. 1, switchover controller 140 controls the manner in which interface modules 130 interact with forwarding planes 120A and 120B based on the status of the forwarding planes. Switchover controller 140 may be implemented in hardware, software, or a combination of hardware and software.

[0022] The operation of switchover controller 140 is illustrated in the flow chart shown in Fig. 3. Switchover controller 140 designates one of the forwarding planes to be the primary forwarding plane (step 300). This designation may be done by default (e.g., initially always selecting forwarding plane 120A) or after polling the forwarding planes to determine a priority. The forwarding planes that are not designated as the primary forwarding plane may be designated as secondary forwarding planes (as described herein, all non-primary forwarding planes are referred to as secondary forwarding planes, though switchover controller 140 does not have to actually designate those planes as such). When interface module 130 is implemented as shown in Fig. 2A, switchover controller 140 enables transmission of packets received at interface modules 130 to be sent to the designated primary forwarding plane (step 310). When interface module 130 is implemented as shown in Fig. 2B, interface modules 130 send received packets to two or more (or all) of the forwarding planes (step 310). The forwarding plane(s) obtain state information from some or all of the received packets, such as data packets (e.g., packet headers), control packets (e.g., routing engine configuration or control traffic), or both.

[0023] Switchover controller 140 enables interface module 130 to transmit an outgoing data stream of packets using packets received from the primary forwarding plane (step 320). By doing so, switchover controller 140 may cause packets received from the secondary forwarding planes to be discarded at interface modules 130. In an alternative embodiment, switchover controller 140 could cause packets from the secondary forwarding planes to be discarded at the secondary forwarding planes before they are received at interface modules 130.

[0024] Switchover controller 140 receives signals from one or more of the forwarding planes and determines the status of each forwarding plane. If switchover controller 140 determines that the primary forwarding plane has failed, it designates one of the secondary forwarding planes to be the primary forwarding plane (step 330). In carrying out this step, switchover controller 140 may first check the status of the secondary forwarding plane to ensure that it is operational before designating it to be the primary forwarding plane. Once the primary forwarding plane has been redesignated, switchover controller 140 performs steps 310 and 320 until the newly designated primary forwarding plane fails.

[0025] While Figs. 1-3 depict one system that incorporates principles of the invention, other systems can also be configured. Fig. 4 shows another such system, system 400, which incorporates principles of the invention. System 400 includes routing engine 410, forwarding plane 420, interface modules 430, service modules 440A and 440B, and switchover controller 450. Forwarding plane 420 is coupled to routing engine 410, interface modules 430, service modules 440A and 440B, and switchover controller 450. Switchover controller is also coupled to service modules 440A and 440B. Like system 100, system 400 is preferably a network switching device (e.g., a router or switch), which interconnects with other switching devices to form part of a network, and accordingly may include other components used in a network switching device.

[0026] Routing engine 410 and forwarding plane 420 are similar to routing engine 110 and one of the forwarding planes 120A and 120B, respectively. Forwarding plane 420 is also

configured to duplicate designated packets and transmit them to two or more outputs (or two or more modules coupled to forwarding plane 420). Forwarding plane 420 may, for example, include multicast functionality that permits it to transmit designated packets to multiple destinations. In a preferred embodiment, forwarding plane 420 can transmit certain packets destined for a service module to both service modules 440A and 440B.

[0027] Interface modules 430 serve as an external interface for system 400. Interface modules 430 connect to respective external lines and forwarding plane 420 to transmit packets between the external lines and forwarding plane 420. In receiving packets, interface modules 430 preferably perform some processing on the received packets (e.g., layer 2/3 header processing) and then transmit the packets to forwarding plane 420. In transmitting packets, interface modules 430 preferably perform some processing from packets received from forwarding plane 420 and then transmit the packets to the respective lines. In a preferred embodiment, interface modules 430 are equipped to handle characteristics unique to the external lines to which they are connected. For example, if the external lines are fiber optic, interface modules 430 may incorporate optical components, such as lasers, to receive and transmit packets over the fiber optic lines. While two interface modules 430 are shown, fewer or more interface modules 430 may be used. In addition, while each interface module 430 is shown as having a single incoming line and a single outgoing line, interface modules 430 could have more incoming and outgoing lines.

[0028] Service modules 440A and 440B are processing modules that connect to forwarding plane 420. Each service module receives packets from forwarding plane 420, processes the received packets in some manner (such as by performing packet encapsulation), and outputs the processed packets back to forwarding plane 420. From the perspective of forwarding plane 420, service modules 440A and 440B may be treated like interface modules 430. Forwarding plane 420 transmits flows of packets to the service modules and receives flows of packets from the service modules. In accordance with the invention, both service modules

receive and process the same packets from forwarding plane 420 (so that they can maintain the same state information), though processed packets from only the primary service module are forwarded by forwarding plane 420.

[0029] Switchover controller 450 controls the manner in which service modules 440A and 44B interact with forwarding plane 420 based on the status of the service modules.

Switchover controller 450 may be implemented in hardware, software, or a combination of hardware and software. The operation of switchover controller 450 is illustrated in the flow chart shown in Fig. 5.

[0030] Switchover controller 450 designates one of the service modules 440A and 440B to be the primary service module (step 500). This designation may be done by default (e.g., initially always selecting service module 440A) or after polling the service modules to determine a priority. The service modules that are not designated as the primary forwarding plane may be designated as secondary service modules (as described herein, all non-primary service modules are referred to as secondary service modules, though switchover controller 140 does not have to designate those modules as such). Switchover controller 450 enables transmission of packets from forwarding plane 420 to be sent to both of the service modules (step 510). Forwarding plane 420 may employ a multicast mechanism so that the same packets are sent to both service modules. In an alternative embodiment, forwarding plane 420 is configured (by switchover controller 450 or in some other manner, such as manually by a user) to send duplicate packets to both service modules. The service modules obtain state information from some or all of the received packets, such as data packets (e.g., packet headers), control packets (e.g., routing engine configuration or control traffic), or both.

[0031] Switchover controller 450 enables packets from the primary service module to be received by forwarding plane 420 for forwarding (step 520). By doing so, switchover controller 450 may cause packets received from the secondary service module to be discarded at the secondary service module or at forwarding plane 420.

[0032] Switchover controller 450 receives signals from one or more of the service modules and determines the status of each service module. If switchover controller 450 determines that the primary service module has failed, it designates one of the secondary service modules to be the primary service module (step 530). In carrying out this step, switchover controller 450 may first check the status of the secondary service module to ensure that it is operational before designating it to be the primary service module. Once the primary service module has been redesignated, switchover controller 450 performs steps 510 and 520 until the newly designated primary service module fails.

[0033] Other embodiments of the invention will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. The invention, for example, may be utilized in any digital system incorporating redundant components whose operation is based on maintained state information. State is maintained in both the primary and secondary components, so that operation can be switched over from the primary to the secondary components without waiting the entire period the secondary component would need to obtain state before it became operational. In the context of a network device, another example of such a component could be a routing engine. Moreover, while state information may be communicated to the primary and secondary components over data traffic, control traffic, or both, state information may also be communicated in other forms and manners. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the invention being indicated by the following claims.